

---

---

# Introducción a Ruby on Rails

— Bundler —

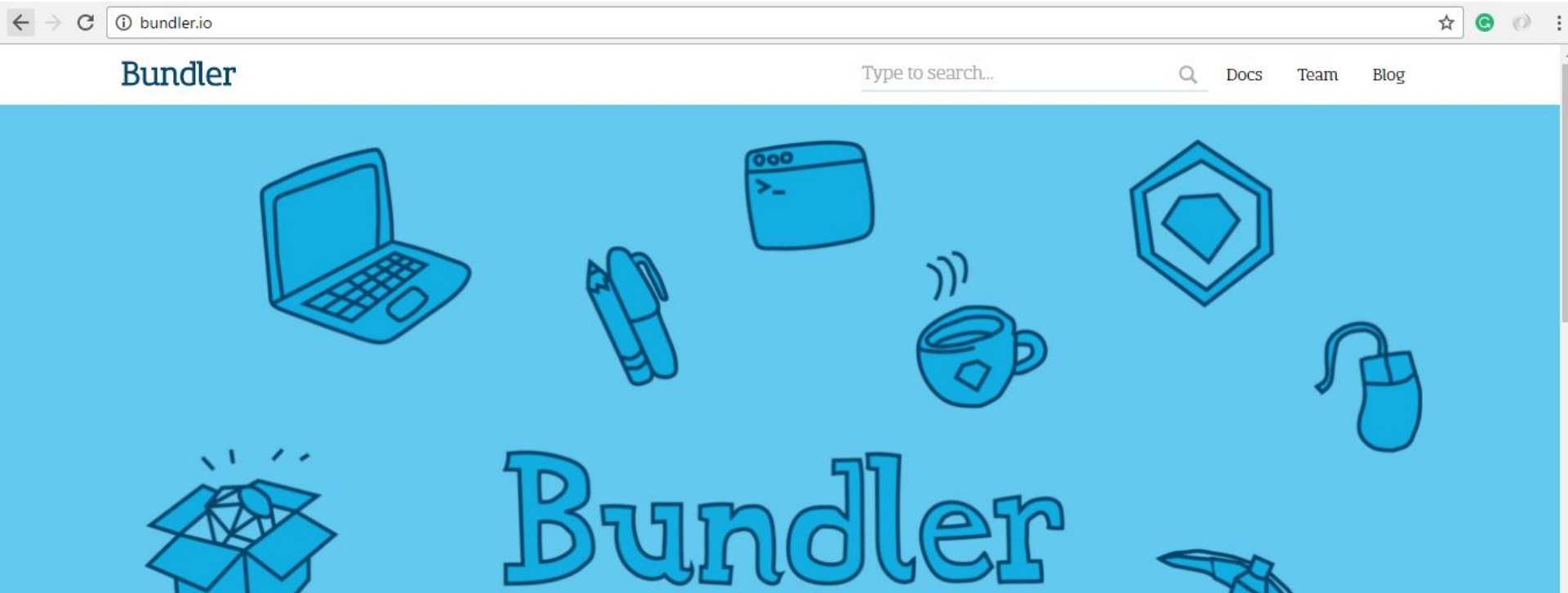
---

---

# Overview

- Bundler
- Gestionando gemas en un proyecto Rails

# Bundler

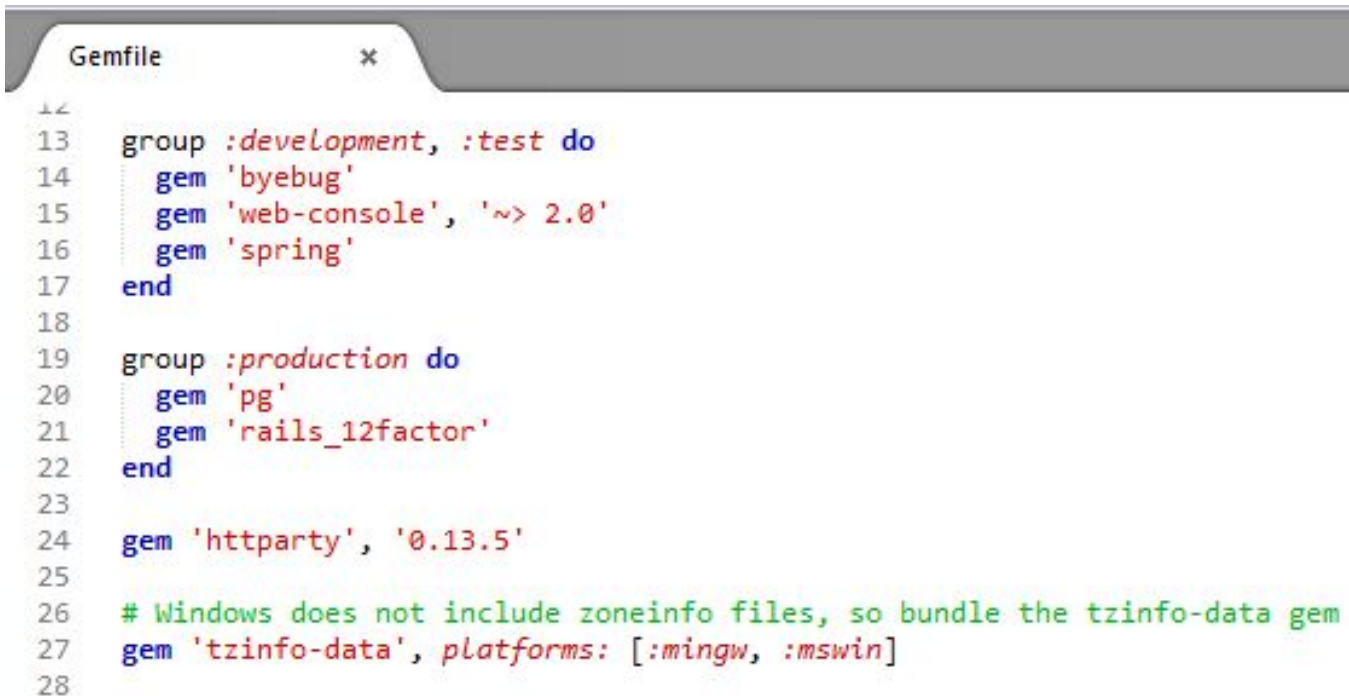


# Bundler

- Permite especificar los gems (y los gems asociados) para la aplicación rails en el archivo Gemfile (en la raíz del proyecto)
- Es la forma ideal de manejar las dependencias en Rails
- Ejecutar bundle install o simplemente bundle después de especificar un nuevo gem en el Gemfile.
- Ejecutar bundle update cuando se modifica la versión de un gem.

# Bundler

Se puede indicar a Rails (mediante el Gemfile) que solamente cargue ciertas gemas dependiendo del “ambiente” o environment.

A screenshot of a code editor window titled 'Gemfile' with a close button 'x'. The code is written in Ruby and defines two groups of gems: :development and :production. The :development group includes 'byebug', 'web-console' (version ~> 2.0), and 'spring'. The :production group includes 'pg' and 'rails\_12factor'. Additionally, there is a standalone gem 'httparty' (version 0.13.5) and a comment about bundling 'tzinfo-data' for Windows platforms (mingw, mswin).

```
12
13 group :development, :test do
14   gem 'byebug'
15   gem 'web-console', '~> 2.0'
16   gem 'spring'
17 end
18
19 group :production do
20   gem 'pg'
21   gem 'rails_12factor'
22 end
23
24 gem 'httparty', '0.13.5'
25
26 # Windows does not include zoneinfo files, so bundle the tzinfo-data gem
27 gem 'tzinfo-data', platforms: [:mingw, :mswin]
28
```

# Que versión se instala?

- Si no se especifica la versión.. se instala la última
- Se puede especificar la versión exacta o una versión aproximada.

```
gem "nokogiri"
```

```
gem "rails", "3.0.0.beta3"
```

```
gem "rack", ">=1.0"
```

```
gem "thin", ">= 1.1", "< 2.0"
```

```
gem "thin", "~>1.1"
```

~> Restricción de versión pesimista.

Se olvida el último dígito y se va aumentando hasta llegar al límite máximo.

# Bundler: require

- Ocasionamente el nombre de la **gema** a ser utilizada en una sentencia **require** es diferente al **nombre** de la **gema**.
- ```
gem 'sqlite3-ruby', require: 'sqlite3'
```

# Ejemplo Gemfile

```
source 'http://rubygems.org'
gem 'rails', '4.2.3'
# Bundle edge Rails instead:
# gem 'rails', github:'git://github.com/rails/rails.git'
gem 'sqlite3'
```

- Nuestra aplicación puede utilizar hasta una versión diferente de **rails** si cambiamos la versión de rails en el **Gemfile** y corremos **bundle update**.
- El **Bundler** crea un archivo **Gemfile.lock**, que contiene las gemas con sus versiones instaladas actualmente y también sus dependencias.



# Bundle exec

Utiliza el ejecutable del gem que está en el gemfile independientemente de cual gem esté instalado.

bundle exec [!\[\]\(919a2cb85b99741a73c0c31a427236a8\_img.jpg\)](#)

`bundle-exec` - Execute a command in the context of the bundle

```
bundle exec [--keep-file-descriptors] command
```

## Description

This command executes the command, making all gems specified in the `Gemfile(5)` available to `require` in Ruby programs.

Essentially, if you would normally have run something like `rspec spec/my_spec.rb`, and you want to use the gems specified in the `Gemfile(5)` and installed via `bundle install(1)`, you should run `bundle exec rspec spec/my_spec.rb`.

Note that `bundle exec` does not require that an executable is available on your shell's `$PATH`.

# Entonces..

- Bundler gestiona las dependencias en un proyecto rails
- Carga las dependencias en el deploy de la aplicación

---> Integración HTTParty con Rails